IA-64 Architecture Innovations

Abbreviated Version of 2/23/99 IA-64 Architecture Disclosure

intel

HEWLETT PACKARD

Agenda

- Review content disclosed at IDF
  - Focus on benefits
  - Focus on “What’s New?”
- Branch Handling: Predication and Prediction
- Speculation
- Register Rotation & Loop Handling

intel

HEWLETT PACKARD

Branch Handling

Traditional Arch

IA-64

Dynamic Prediction

Static Prediction

Predication

& Others

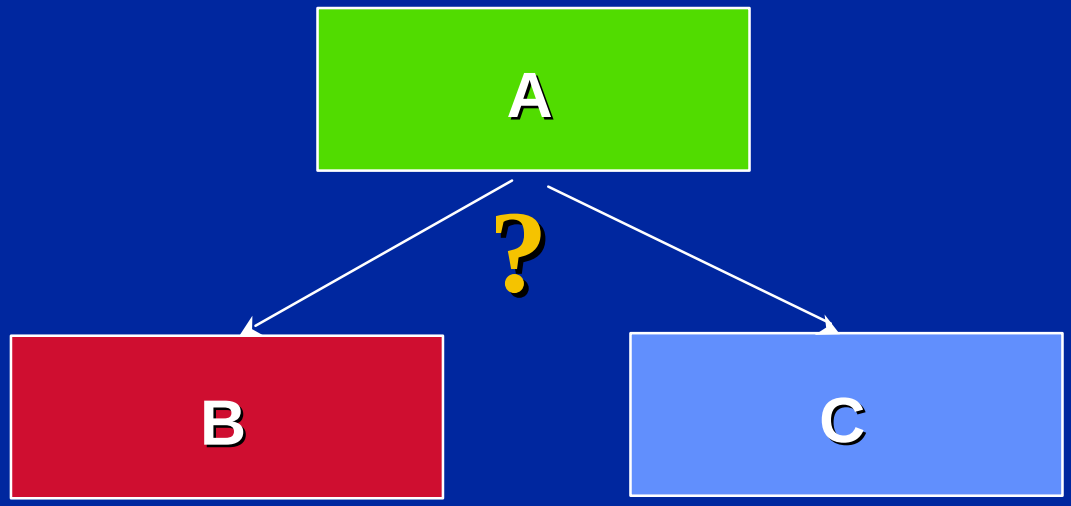
- Branches are forks in code
- Can indicate a decision
- Common in wide variety of applications

intel

HEWLETT PACKARD

Review

Dynamic Branch Prediction



- Guesses either B or C
- Suffers performance penalty when mispredicted

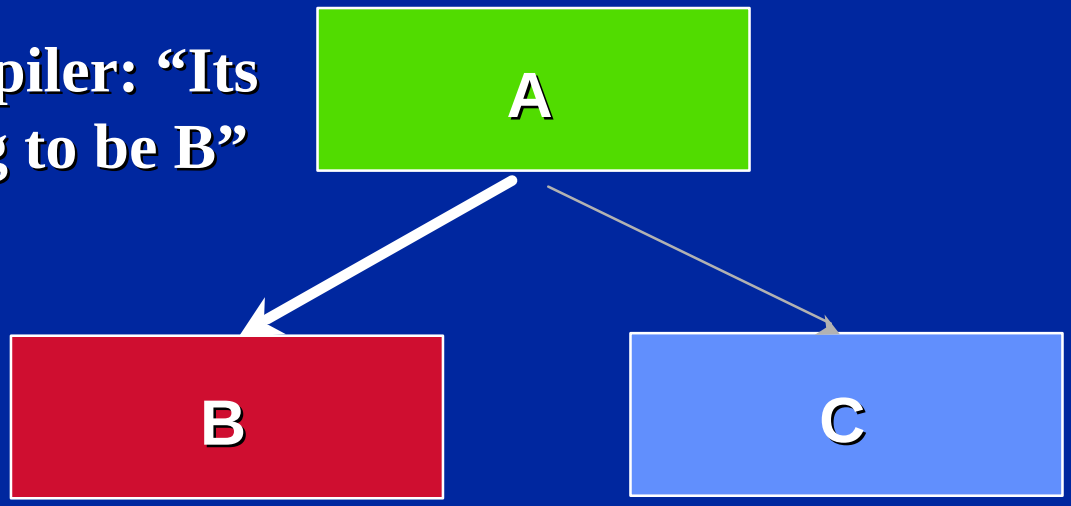
intel

HEWLETT PACKARD

New!

Static Branch Prediction

Compiler: “Its going to be B”

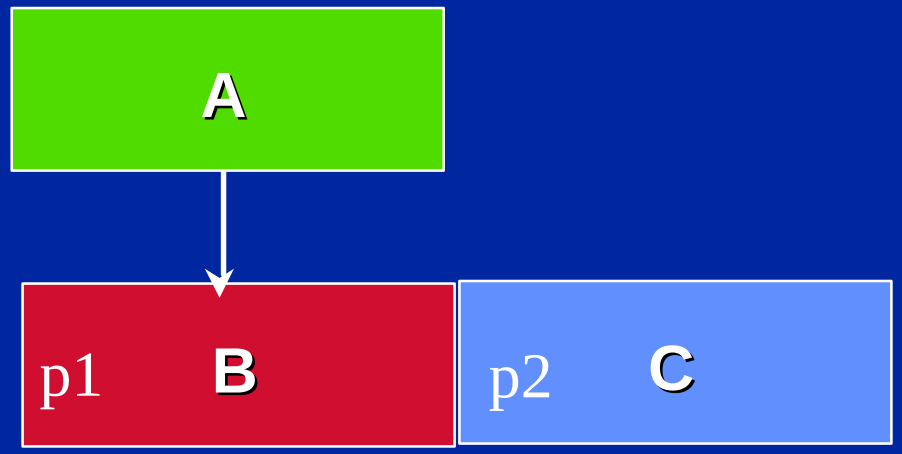


- Compiler knows which one is almost always taken or never taken
- Removes guesswork for processor, reduces mispredict penalties

intel

HEWLETT PACKARD

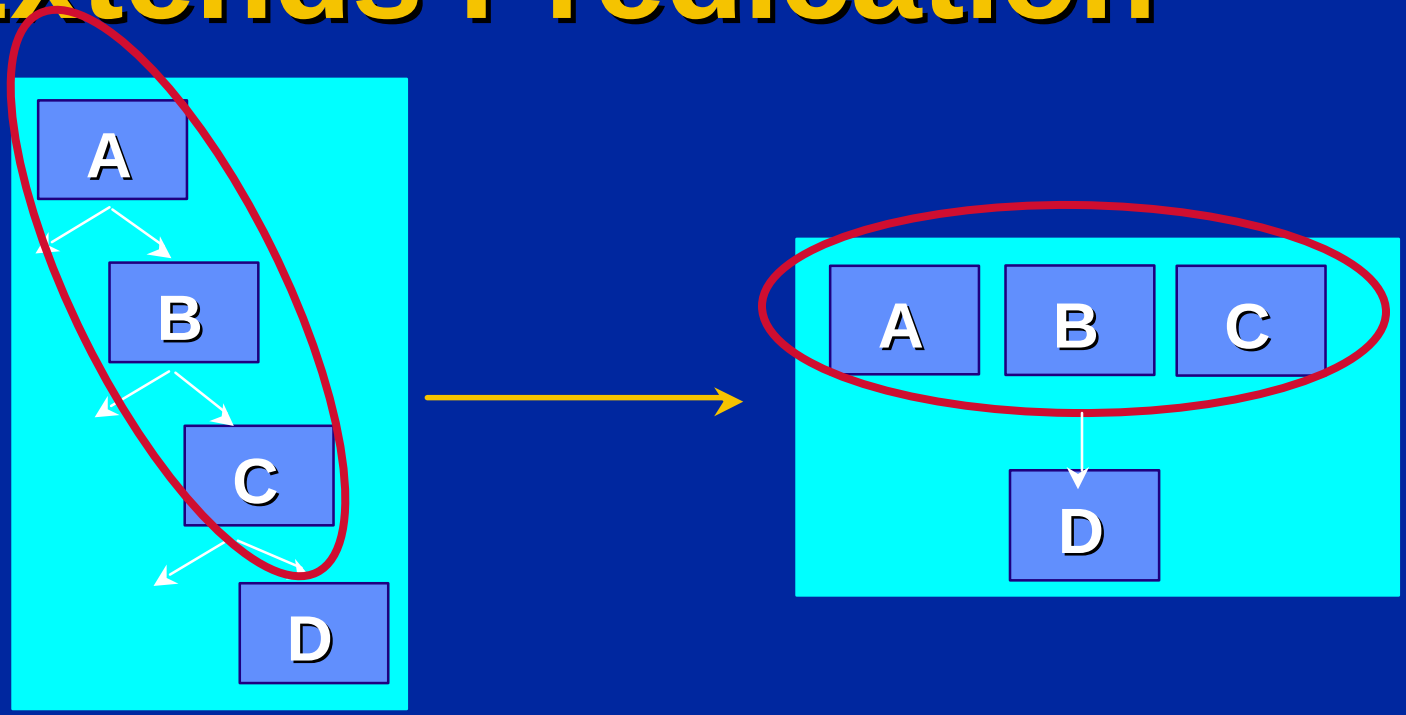
Predication



- Removes Branch, executes B&C in parallel
- Avoids possibility of mispredict
- Predicates (p1 & p2) are bits that turn on/off B & C
- Biggest benefit to code w/ hard to predict branches
  - Large server apps
  - Data sorting

New!

Parallel Compares: Extends Predication



Enables reduction from 7 to 5 cycles on queens loop

Reduces critical path, further increasing performance

intel

HEWLETT PACKARD

Predication Benefits

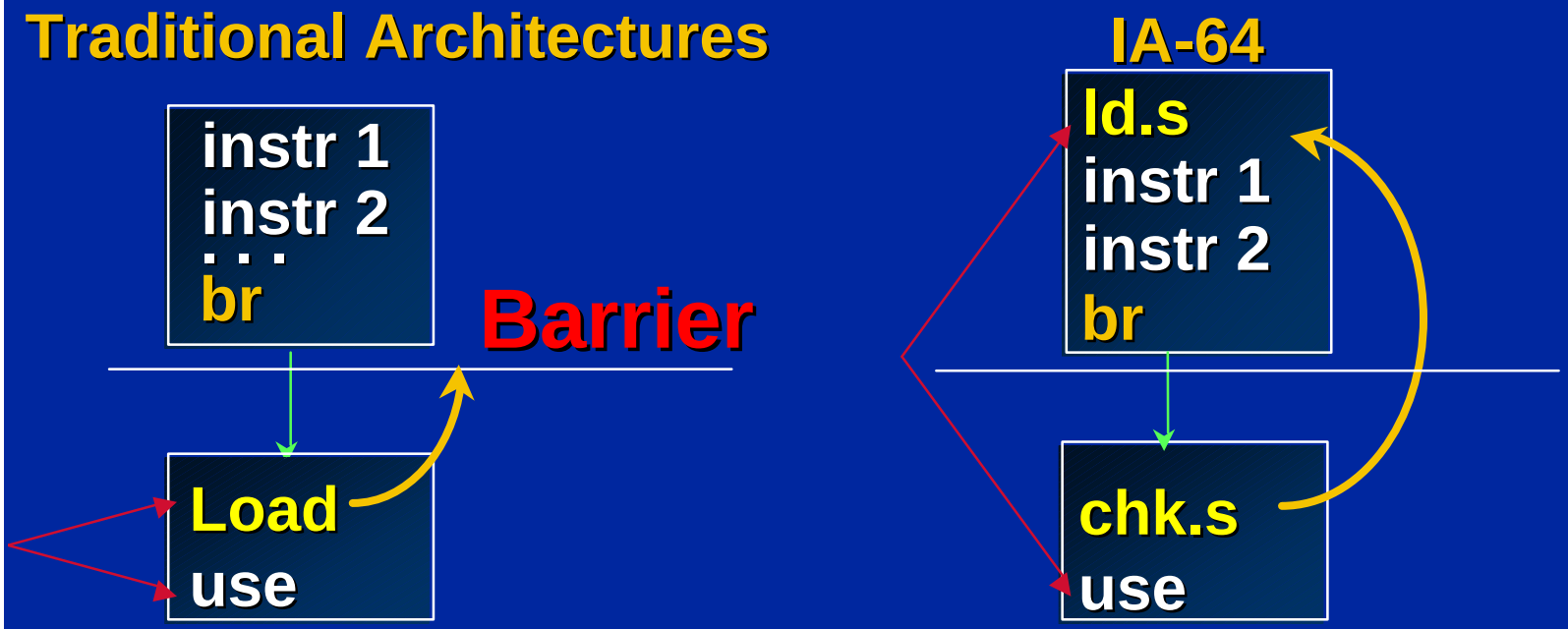
- Reduces branches and mispredict penalties
  - 50% fewer branches and 37% faster code*
- Parallel compares further reduce critical paths
- Greatly improves code with hard to predict branches
  - Large server apps- capacity limited
  - Sorting, data mining- large database apps
  - Data compression
- Traditional architectures' "bolt-on" approach can't efficiently approximate predication
  - Cmove: 39% more instructions, 30% lower performance*
  - Instructions must all be speculative

intel

HEWLETT PACKARD

* Source: S. Mahlke, 1995

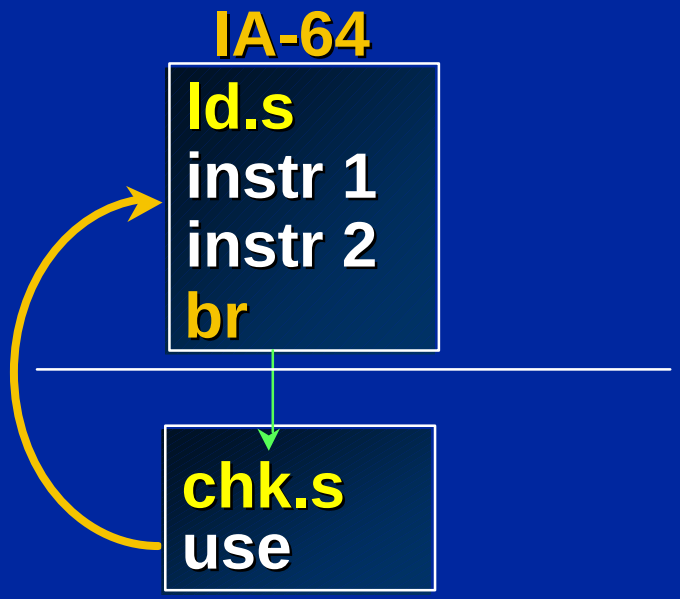
Speculation Review



- Moving things up enables them to be performed sooner
- Loads take time, need to separate from use
  - Memory latency is a major problem in today's systems
  - CPUs getting faster than memory

New!

Hoisting Uses



- The uses of speculative data can also be executed speculatively
- Provides additional scheduling flexibility to achieve greater parallelism

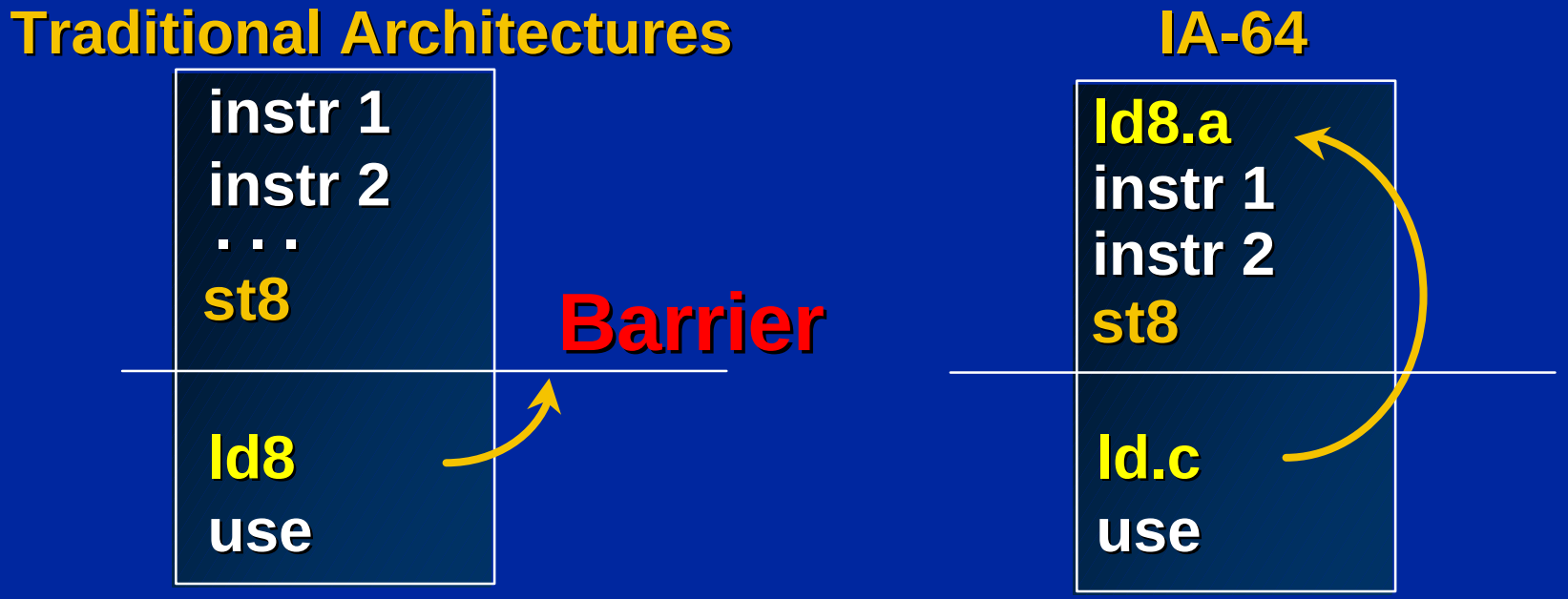
intel

HEWLETT PACKARD

New!

Introducing Data Speculation

- Compiler can issue a load prior to a preceding, possibly-conflicting store

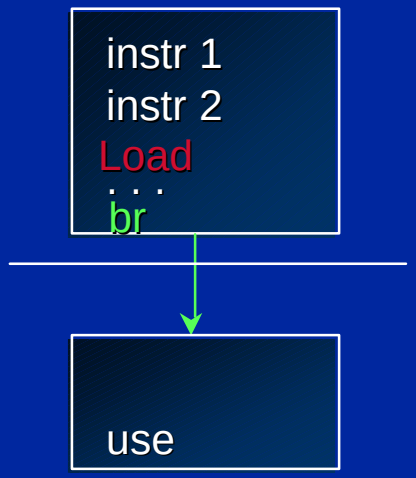


Unique feature to IA-64

New!

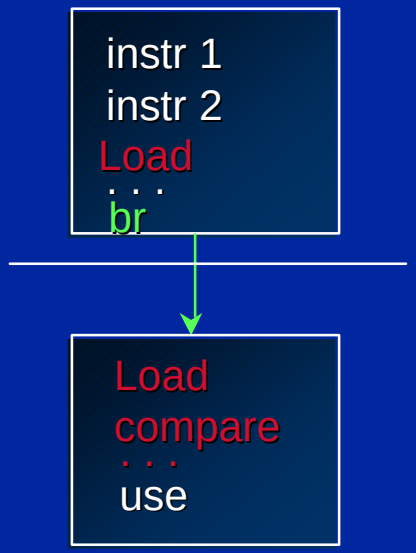
Exception Handling

Traditional Method 1



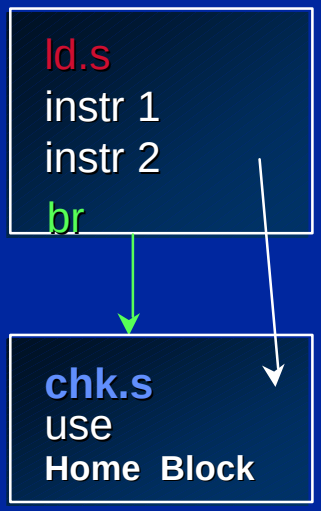
Only when you know it's safe

Traditional Method 2



Extra baggage to ensure integrity

IA-64



;Exception Detection

NaT Bit

;Exception Delivery

NaT bit ensures integrity of data without penalty

intel

HEWLETT PACKARD

Speculation Benefits

- Reduces impact of memory latency
  - Performance improvement at 79% when combined with predication*
- Greatest improvement to code with many cache accesses
  - Large databases
  - Operating systems
- Scheduling flexibility enables new levels of performance headroom

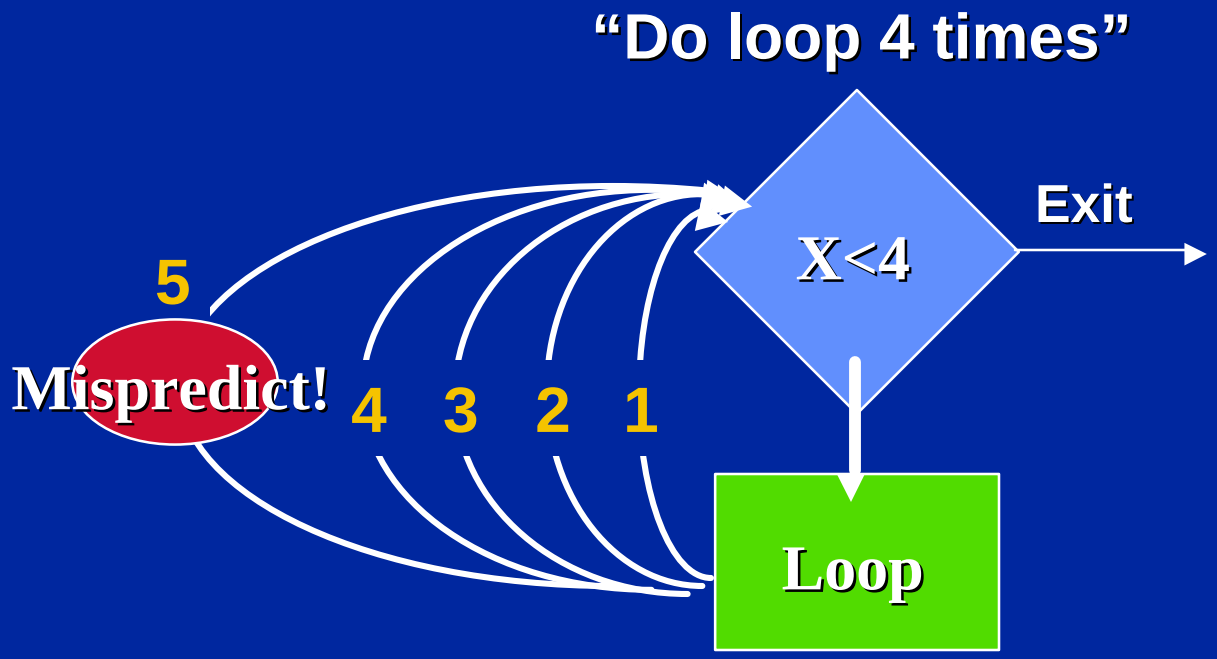
intel

* August, 1998

HEWLETT PACKARD

New!

Loop Handling



IA-64 has special register to avoid #5: called Loop Counter (LC)

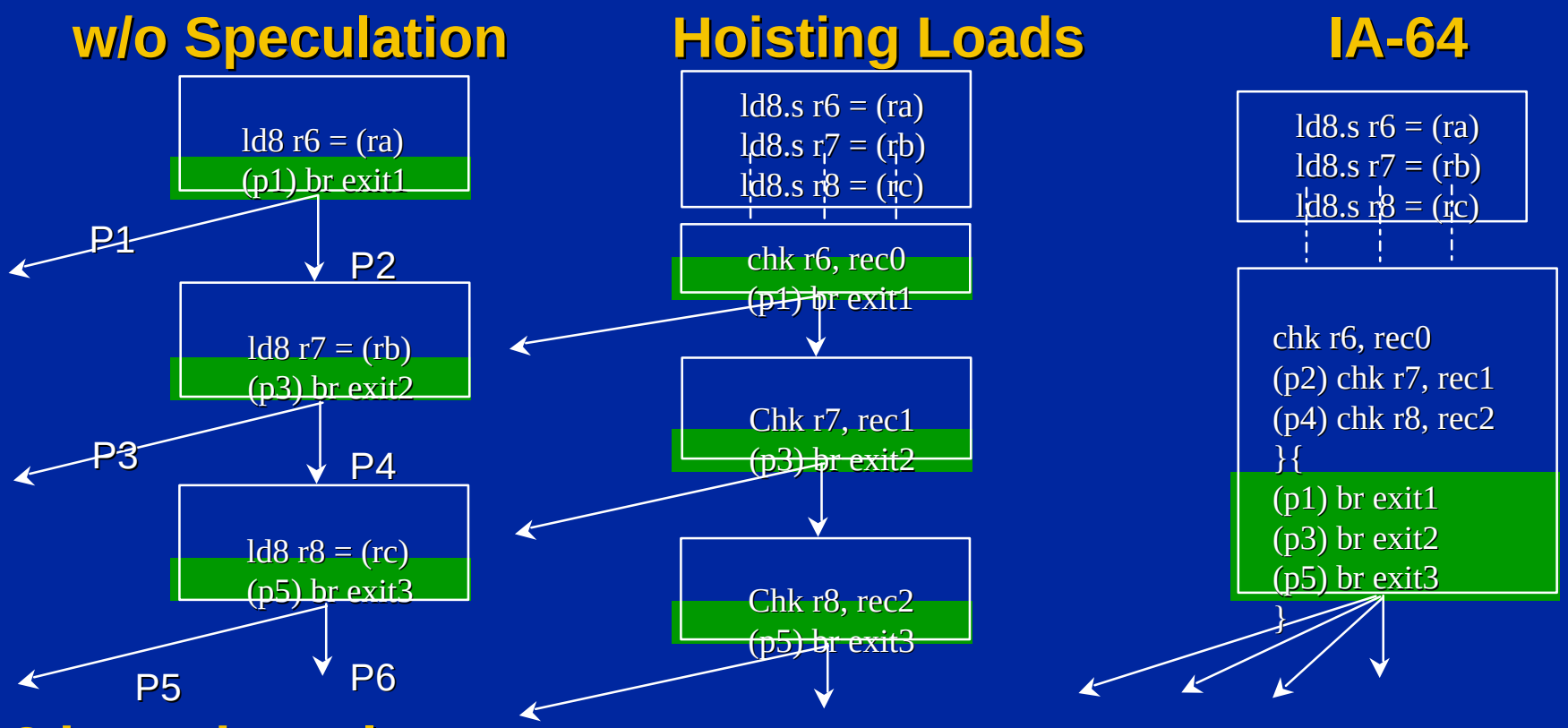
Improves integer code performance

intel

HEWLETT PACKARD

New!

Multi-way Branch



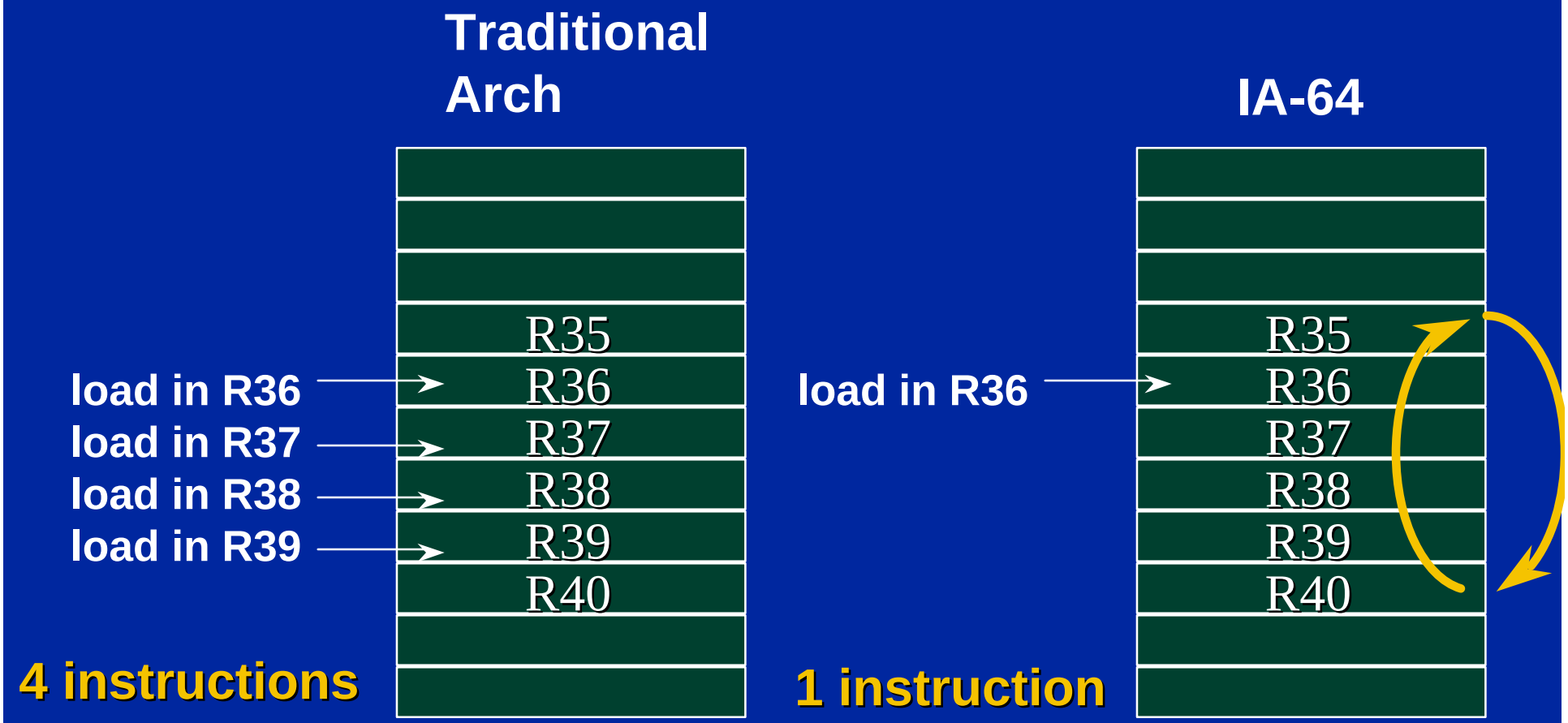
- Multiway branches: more than 1 branch in a single cycle
  - Chaining multiway branches allows n-way branching

intel

HEWLETT PACKARD

New!

Introducing Rotating Registers



IA-64 Registers rotate, load stays the same

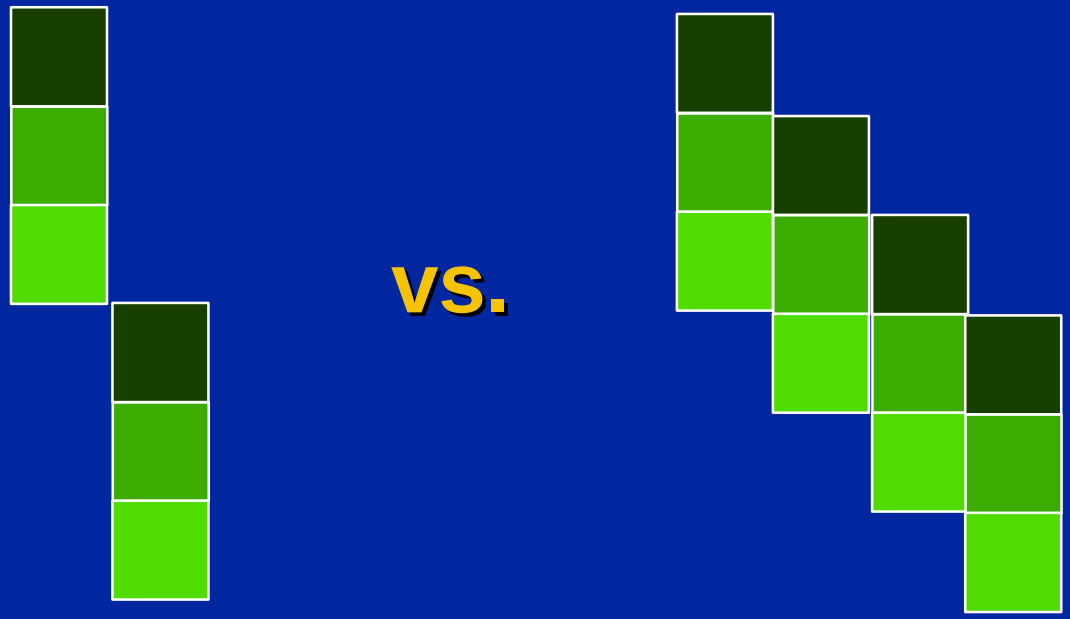
intel

HEWLETT PACKARD

New!

Software Pipelining

- Overlapping execution of different loop iterations



- More iterations in same amount of time

IA-64 Offers a Unique Approach

intel

HEWLETT PACKARD

Software Pipelining Benefits

- Loop pipelining maximizes performance; minimizes overhead
  - Avoids code expansion of unrolling and code explosion of prologue and epilogue
  - Smaller code means fewer cache misses
  - Greater performance improvements in higher latency conditions
- Reduced overhead allows S/W pipelining of small loops with unknown trip counts
  - Typical of integer scalar codes

intel

HEWLETT PACKARD

Reviewing What's New:

- Parallel compares
- Tbit
- Nat bits
- Deferral
- Hoisting uses
- Propagation
- Branch instructions
- Static prediction
- Loop branches
- LC register
- EC register
- Multiway branch
- Branch registers
- Register rotation
- Predicate rotation
- RRBs

intel

HEWLETT PACKARD

Feature Comparison

Traditional Architectures	IA-64
Dynamic branch prediction	Static Prediction and Predication enhance dynamic prediction to reduce mispredict penalties
Conditional moves limited in applicability and require additional instructions	Predication widely applicable, parallel compares further enhance benefit
Non faulting loads limited to certain conditions or require additional instructions	Control and data speculation enable greater scheduling freedom of loads
Software pipelining limited to large loops due to code size explosion	Rotating registers and rotating predicates allow wide application of software pipelining performance benefits without the code growth

intel

HEWLETT PACKARD

Summary

- Speculation reduces memory latency
  - IA-64 removes recovery from critical path
  - Good for variety of server applications
- Predication removes branches
  - Logical and / or of compares increases parallelism
  - Good for large databases applications
- S/W pipelining support with minimal overhead enables broad usage
  - Performance for small integer loops with unknown trip counts as well as monster FP loops

intel

HEWLETT PACKARD